GUNADARMA UNIVERSITY

A/B Testing Mekargo.id Website of Mekar PT Sampoerna Wirausaha with Bayesian Inference and Pymc3

ALVIAN DWI KURNIANTO

92216007

BUSINESS INFORMATION SYSTEM

SUPERVISOR: DR. Setia Wirawan

UG | PENULISANILMIAH UNIVERSITASGUNADARMA

OUTLINE

INTRODUCTION

- Background
- Problem Identification
- Scope of Problem
- Research Purposes
- Research Benefits

RESULT AND DISCUSSION

- Identifying Problems
- Defining Website Measurement
- Developing Hypothesis
- Developing & Testing Page Variation
- Analyzing Test Results

CONCLUTION AND SUGGESTION

- Conclusion
- Suggestion

BACKGROUND

- Small Medium Enterprise (SMEs) has an important role in encouraging the growth of the Indonesian economy.
- Currently, SMEs have been the mainagenda of Indonesia's economic development
- The most prominent issue of SMEs involves providing business financing or business capital
- However, to partner with banks, small businesses are required to present business proposals that are feasible or feasible and profitable
- Mekar provides solutions to SMEs and other Consumers to get Financial Services Access or Capital in Indonesia.

PROBLEM IDENTIFICATION

The problem discussed in this research is to determine which version of mekargo.id website at PT Mekar Sampoerna Wirausaha with A / B testing which become the best version with bayesian inference method with pymc3 tools.

SCOPE OF PROBLEM

1. The object of research is the website mekargo.id.
2. The research was conducted at PT Mekar Sampoerna Wirausaha.
3. Study time between August 2017 to January 2018.
4. Data collection method for A / B testing techniques on mekargo.id website with the help of Google Analytics tools.
5. The generated data from A / B testing is analyzed by Bayesian inference analysis using pymc3, numpy, matplotlib, scipy, jupyter and Ipython tools with python programming language.
6. Mekargo.id website features that studied is a feature of input data by borrower.
7. Mekargo.id website feature that is not discussed is a static page feature that contains the home page, about us, terms and conditions, privacy policy and contact us.

RESEARCH PURPOSES

The purpose of this research is to know the version of A / B testing which is the best version of mekargo.id website at PT Mekar Sampoerna Entrepreneurship with bayesian inference method with pymc3.

RESEARCH BENEFITS

The benefits of this research are:

1. For the researcher, be a guide or research reference in the field of information system in A / B testing with bayessian inference method.
2. For other researchers, as a literature source for A / B testing withbayessian inference method.
3. For the company under study, provide an alternative to solving the problem of choosing the best version with A / B testing.

IDENTIFYING PROBLEMS

PT Mekar or Sampoerna Wirausaha is a fintech company running a busi ness with a peer-to-peer loan platform. Mekar Go is a website used to collect borrower data. The data collected on the current version of Mekar Go web- site shows the conversion of a borrower who completes data filling from start to finish less than 10%. This conversion percentage by the marketing team is too small and needs to be improved.

DEFINING WEB MEASUREMENT

1. Business Objective

   The desired business objective is to increase the number of borrowers that will be given a peer-to-peer lending loan.

2. Website Goal

   Increase the number of borrowers who register, increase the percentage completion of filling the registration form by the borrower.

3. Key Performance Metric

   The number of registrants per month and the percentage completion of form filling by the borrower upon filling in the registration form

4. Target Metric

   The expected target is 500 unique visitors who register for each version and percentage of data completion above 10% for each version

Gunadarma UG University

DEVELOPING & TEST PAGE VARIANTS

Developing Page Variants

Table 4.1: Data fields changes

Part	Fields Name	Remove	Optional	New
Personal Detail Information	Alamat Usaha / Tempat Bekerja (Provinsi, Kota, Kecamatan, Kelurahan)	v		
	Bidang Jenis Usaha / Jenis Pekerjaan	v		
	Lama Usaha / Lama Bekerja	v		
	Jumlah Karyawan	v		
	Foto Tempat Usaha	v		
	Status Karyawan	v		
	Email		v	
	Foto KTP		v	
	Tanggal Lahir			v
Jaminan	Foto Tanah + Bangunan		v	
	Foto Kendaraan Bermotor		v	

Gunadarma UG University

DEVELOPING & TEST PAGE VARIANTS

Developing Page Variants

Table 4.2: Url definition

Step	Site A (/ukm/a/)	Site B (/ukm/b/)	Site C (/ukm/c/)
Survey Needs Page	/ukm/a/survei	/ukm/b/survei	/ukm/c/survei
Detail Needs Page	/ukm/a/detil-survei	/ukm/b/detil-survei	/ukm/c/detil-survei
Personal Information Page	/ukm/a/data-diri	/ukm/b/data-diri	/ukm/c/data-diri
Thanks Page	/ukm/a/terima-kasih	/ukm/b/terima-kasih	/ukm/c/terima-kasih
Detail Loan Access Page	/ukm/a/ pinjaman	/ukm/b/ pinjaman	/ukm/c/ pinjaman
Ekstra Pesonal Detail Information Page	/ukm/a/ tambahan-data-diri	-	-
Congrats Page	/ukm/a/selamat	/ukm/b/selamat	/ukm/c/selamat

Gunadarma UG University

RESULT AND DISCUSSION

Gunadarma UG University

DEVELOPING & TEST PAGE VARIANTS

Developing Page Variants

Listing 4.1: Traffic splitting

```
class RoundRobin(models.Model):
    flow_type = models.CharField(max_length=1, null=True)
    partner_slug = models.SlugField(null=True)
    def next(self):
        latest = self.flow_type
        if latest == 'a':
            return 'b'
        elif latest == 'b':
            return 'c'
        elif latest == 'c':
            return 'a'

if RoundRobin.objects.count() == 0:
    next_flow = 'a'
else:
    latest = RoundRobin.objects.last()
    next_flow = latest.next()
```

Gunadarma UG University

DEVELOPING & TEST PAGE VARIANTS

Testing Page Variants

Listing 4.2: Google Analytics implementation

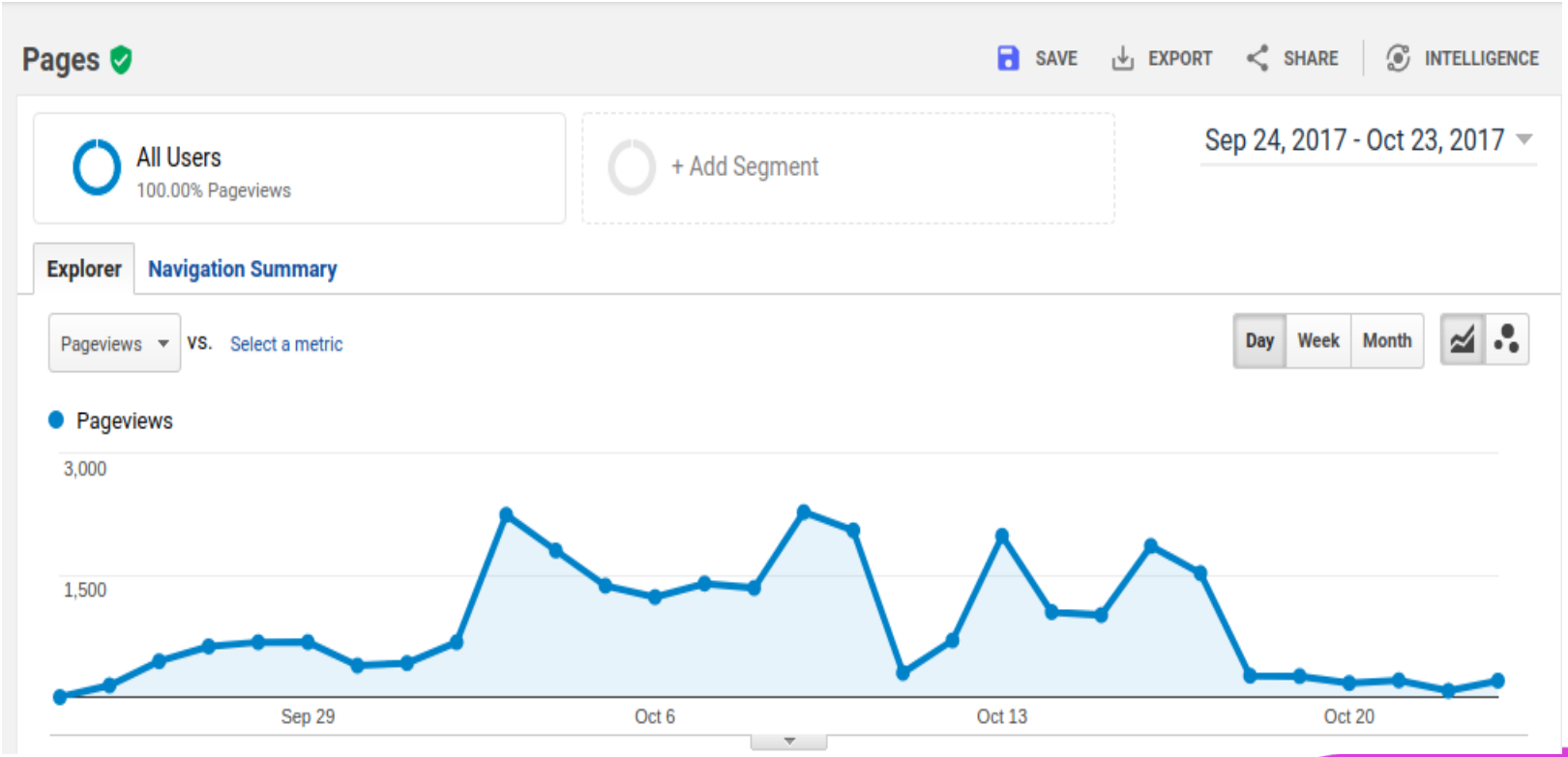
```
1 <!-- Global Site Tag (gtag.js) - Google Analytics -->
2 <script async src="https://www.googletagmanager.com/gtag/
    js?id=UA-70057602-1">
3 </script>
4 <script>
5   window.dataLayer = window.dataLayer || [];
6   function gtag(){dataLayer.push(arguments);}
7   gtag('js', new Date());
8   gtag('config', 'UA-70057602-1');
9 </script>
```

RESULT AND DISCUSSION

Gunadarma UG University

DEVELOPING & TEST PAGE VARIANTS

Testing Page Variants



RESULT AND DISCUSSION

Gunadarma University

DEVELOPING & TEST PAGE VARIANTS

Testing Page Variants

Primary Dimension: **Page** Page Title Other

Plot Rows | Secondary dimension | Sort Type: Default | /ukm/ | advanced

	Page	Pageviews	Unique Pageviews	Avg. Time on Page	Entrances	Bounce Rate	% Exit	Page Value
		27,335 % of Total: 97.94% (27,911)	19,711 % of Total: 97.72% (20,170)	00:01:23 Avg for View: 00:01:23 (0.13%)	10,343 % of Total: 98.15% (10,538)	54.08% Avg for View: 53.77% (0.59%)	37.85% Avg for View: 37.76% (0.25%)	$0.00 % of Total: 0.00% ($0.00)
1.	/ukm/c/	5,164 (18.89%)	**3,801** (19.28%)	00:00:55	3,709 (35.86%)	56.70%	56.66%	$0.00 (0.00%)
2.	/ukm/b/	4,737 (17.33%)	**3,385** (17.17%)	00:01:02	3,281 (31.72%)	52.45%	51.61%	$0.00 (0.00%)
3.	/ukm/a/	4,222 (15.45%)	**3,123** (15.84%)	00:00:52	3,039 (29.38%)	52.45%	50.76%	$0.00 (0.00%)
4.	/ukm/b/data-diri/	2,200 (8.05%)	**1,497** (7.59%)	00:03:30	76 (0.73%)	51.32%	30.32%	$0.00 (0.00%)
5.	/ukm/c/pinjaman/	1,894 (6.93%)	**1,494** (7.58%)	00:01:35	35 (0.34%)	71.43%	38.01%	$0.00 (0.00%)
6.	/ukm/a/survei/	2,270 (8.30%)	**1,392** (7.06%)	00:00:37	44 (0.43%)	54.55%	10.66%	$0.00 (0.00%)
7.	/ukm/a/detil-survei/	1,913 (7.00%)	**1,156** (5.86%)	00:00:31	24 (0.23%)	50.00%	6.33%	$0.00 (0.00%)
8.	/ukm/a/data-diri/	1,725 (6.31%)	**1,127** (5.72%)	00:02:30	22 (0.21%)	59.09%	14.38%	$0.00 (0.00%)
9.	/ukm/a/pinjaman/	838 (3.07%)	**751** (3.81%)	00:02:14	22 (0.21%)	68.18%	30.91%	$0.00 (0.00%)
10.	/ukm/b/pinjaman/	569 (2.08%)	**501** (2.54%)	00:02:21	26 (0.25%)	42.31%	36.20%	$0.00 (0.00%)

Show rows: 10 Go to: 1 1 - 10 of 21

This report was generated on 1/31/18 at 7:05:00 PM - Refresh Report

Gunadarma UG University

Analyzing Test Results

In [1]:
```
import pymc3 as pm
import numpy as np
%matplotlib inline
from IPython.core.pylabtools import figsize
import matplotlib.pyplot as plt
import scipy.stats as stats
figsize(12.5, 4)
```

Gunadarma UG University

Analyzing Test Results

In [2]:
```
true_A = 120
true_B = 129
true_C = 84

N_sample = 1156
```

Gunadarma UG University

Analyzing Test Results

In [3]:
```
true_p_A = true_A/float(N_sample)
true_p_B = true_B/float(N_sample)
true_p_C = true_C/float(N_sample)

print("true p_A:", true_p_A)
print("true p_B:", true_p_B)
print("true p_C:", true_p_C)
```

```
true p_A: 0.10380622837370242
true p_B: 0.1115916955017301
true p_C: 0.0726643598615917
```

RESULT AND DISCUSSION

Gunadarma UG University

Analyzing Test Results

In [4]:
```
observations_A = stats.bernoulli.rvs(true_p_A, size=N_sample)
observations_B = stats.bernoulli.rvs(true_p_B, size=N_sample)
observations_C = stats.bernoulli.rvs(true_p_C, size=N_sample)

print("Obs from Site A: ", observations_A[:30], "...")
print("Obs from Site B: ", observations_B[:30], "...")
print("Obs from Site C: ", observations_C[:30], "...")
```

```
Obs from Site A:  [0 0 1 0 0 1 0 0 0 0 0 0 0 0 1 0 0 1 0 0 0 0 0 0 0 0 0 0] ...
Obs from Site B:  [0 1 0 1 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 0 0] ...
Obs from Site C:  [0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 1 0 0 0 0 0 0 0] ...
```

Gunadarma UG University

Analyzing Test Results

In [5]:
```
print(np.mean(observations_A))
print(np.mean(observations_B))
print(np.mean(observations_C))

print(np.sum(observations_A))
print(np.sum(observations_B))
print(np.sum(observations_C))
```

```
0.0951557093426
0.117647058824
0.0726643598616
110
136
84
```

RESULT AND DISCUSSION

Gunadarma University

Analyzing Test Results

In [6]:
```
with pm.Model() as model:
    p_A = pm.Uniform("p_A", 0, 1)
    p_B = pm.Uniform("p_B", 0, 1)
    p_C = pm.Uniform("p_C", 0, 1)

    # Define the deterministic delta function. This is our unknown of interest.
    delta_A_B = pm.Deterministic("delta_A_B", p_A - p_B)
    delta_A_C = pm.Deterministic("delta_A_C", p_A - p_C)
    delta_B_C = pm.Deterministic("delta_B_C", p_B - p_C)

    # Set of observations, in this case we have three observation datasets.
    obs_A = pm.Bernoulli("obs_A", p_A, observed=observations_A)
    obs_B = pm.Bernoulli("obs_B", p_B, observed=observations_B)
    obs_C = pm.Bernoulli("obs_C", p_C, observed=observations_C)

    step = pm.Metropolis()
    trace = pm.sample(20000, step=step)
    burned_trace=trace[1000:]
```

```
100%|██████████| 20500/20500 [00:31<00:00, 643.38it/s]
```

Gunadarma UG University

Analyzing Test Results

```
In [7]: p_A_samples = burned_trace["p_A"]
        p_B_samples = burned_trace["p_B"]
        p_C_samples = burned_trace["p_C"]
        delta_A_B_samples = burned_trace["delta_A_B"]
        delta_A_C_samples = burned_trace["delta_A_C"]
        delta_B_C_samples = burned_trace["delta_B_C"]
```

RESULT AND DISCUSSION

Gunadarma University

Analyzing Test Results

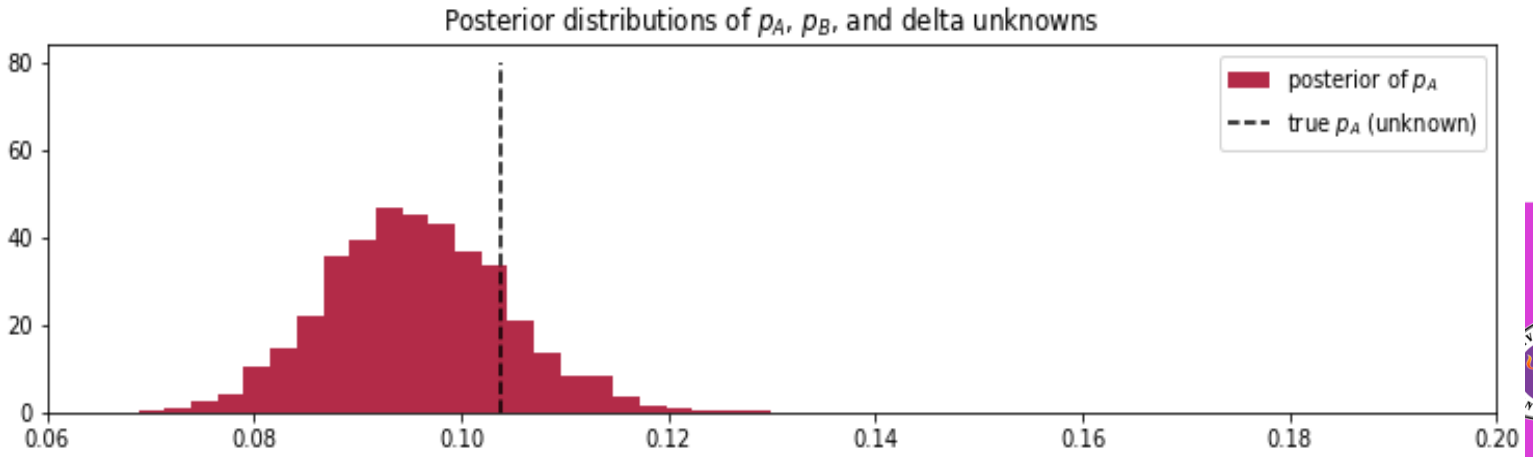
In [8]:
```
figsize(12.5, 10)

#histogram of posteriors

ax = plt.subplot(311)

plt.xlim(.06, .2)
plt.hist(p_A_samples, histtype='stepfilled', bins=25, alpha=0.85,
         label="posterior of $p_A$", color="#A60628", normed=True)
plt.vlines(true_p_A, 0, 80, linestyle="--", label="true $p_A$ (unknown)")
plt.legend(loc="upper right")
plt.title("Posterior distributions of $p_A$, $p_B$, and delta unknowns")
```

Out[8]:
```
Text(0.5,1,'Posterior distributions of $p A$, $p B$, and delta unknowns')
```



RESULT AND DISCUSSION

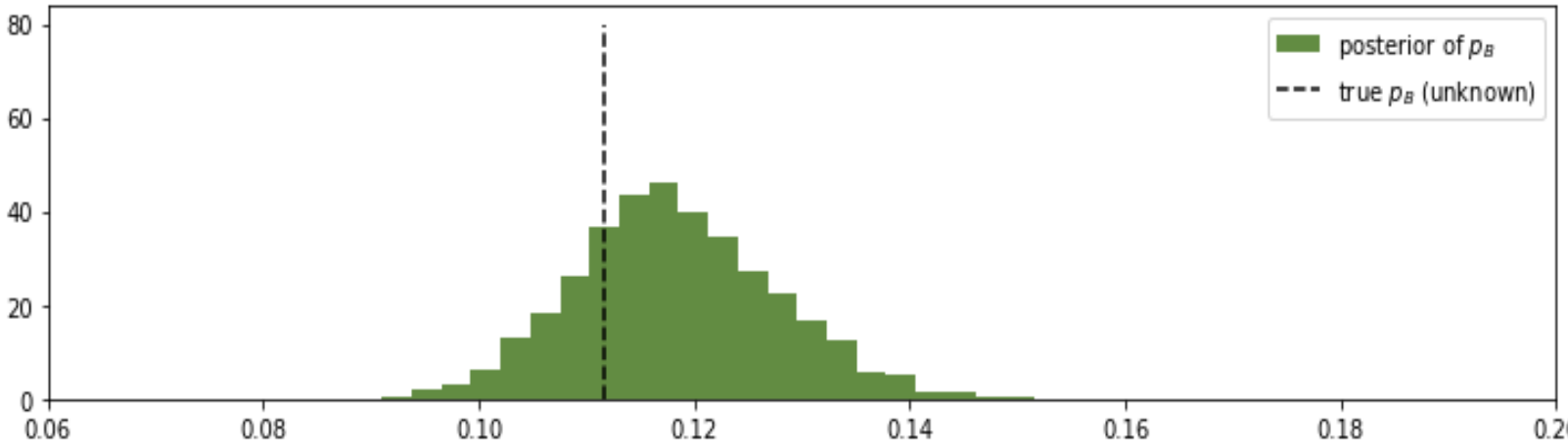
Gunadarma UG University

Analyzing Test Results

In [9]:
```
ax = plt.subplot(312)

plt.xlim(.06, .2)
plt.hist(p_B_samples, histtype='stepfilled', bins=25, alpha=0.85,
         label="posterior of $p_B$", color="#467821", normed=True)
plt.vlines(true_p_B, 0, 80, linestyle="--", label="true $p_B$ (unknown)")
plt.legend(loc="upper right")
```

Out[9]: <matplotlib.legend.Legend at 0x7f03334969e8>



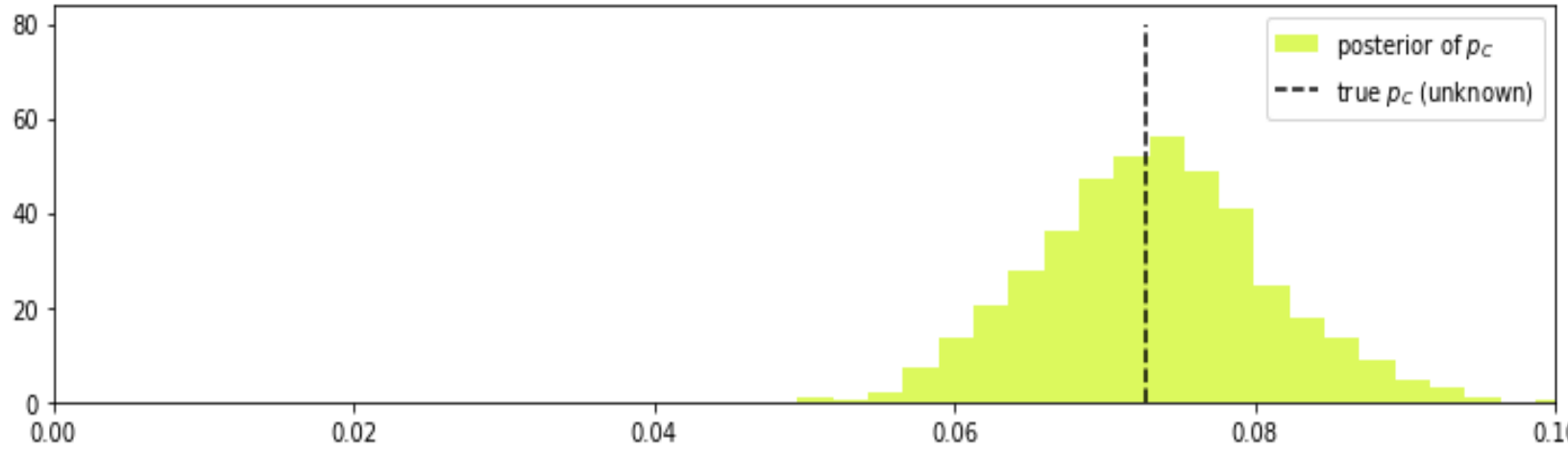
Gunadarma University

Analyzing Test Results

In [10]:
```
ax = plt.subplot(313)

plt.xlim(0, .1)
plt.hist(p_C_samples, histtype='stepfilled', bins=25, alpha=0.85,
         label="posterior of $p_C$", color="#D6F841", normed=True)
plt.vlines(true_p_C, 0, 80, linestyle="--", label="true $p_C$ (unknown)")
plt.legend(loc="upper right")
```

Out[10]:
```
<matplotlib.legend.Legend at 0x7f0332445a90>
```



Gunadarma UG University

Analyzing Test Results

Table 4.3: List of variables of A,B, and C version

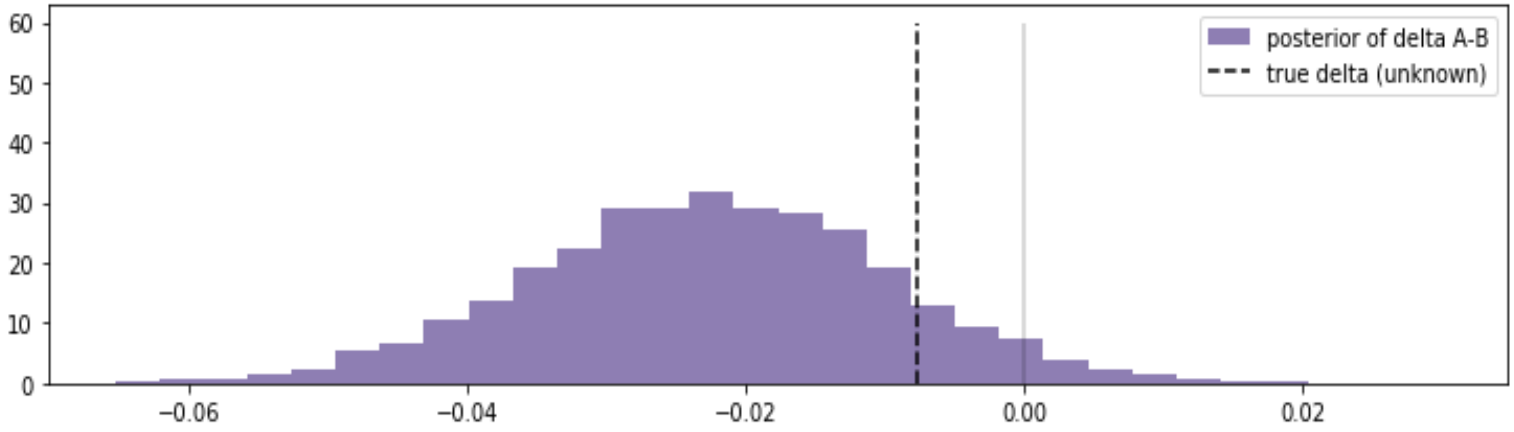
Variable name	Version A	Version B	Version C
users completing the data completion	true_A 120	true_B 129	true_C 84
completion percentage	true_p_A 0.10380622837370242	true_p_B 0.1115916955017301	true_p_C 0.0726643598615917
sum of the True value from bernoulli observations	sum_true_p_A 114	sum_true_p_B 138	sum_true_p_C 86
mean of the bernoulli observations	mean_p_A 0.098615916955	mean_p_B 0.11937716263	mean_p_C 0.0743944636678

RESULT AND DISCUSSION

Gunadarma UG University

Analyzing Test Results

In [11]:
```
ax = plt.subplot(313)
plt.hist(delta_A_B_samples, histtype='stepfilled', bins=30, alpha=0.85,
         label="posterior of delta A-B", color="#7A68A6", normed=True)
plt.vlines(true_p_A - true_p_B, 0, 60, linestyle="--",
           label="true delta (unknown)")
plt.vlines(0, 0, 60, color="black", alpha=0.2)
plt.legend(loc="upper right");
```

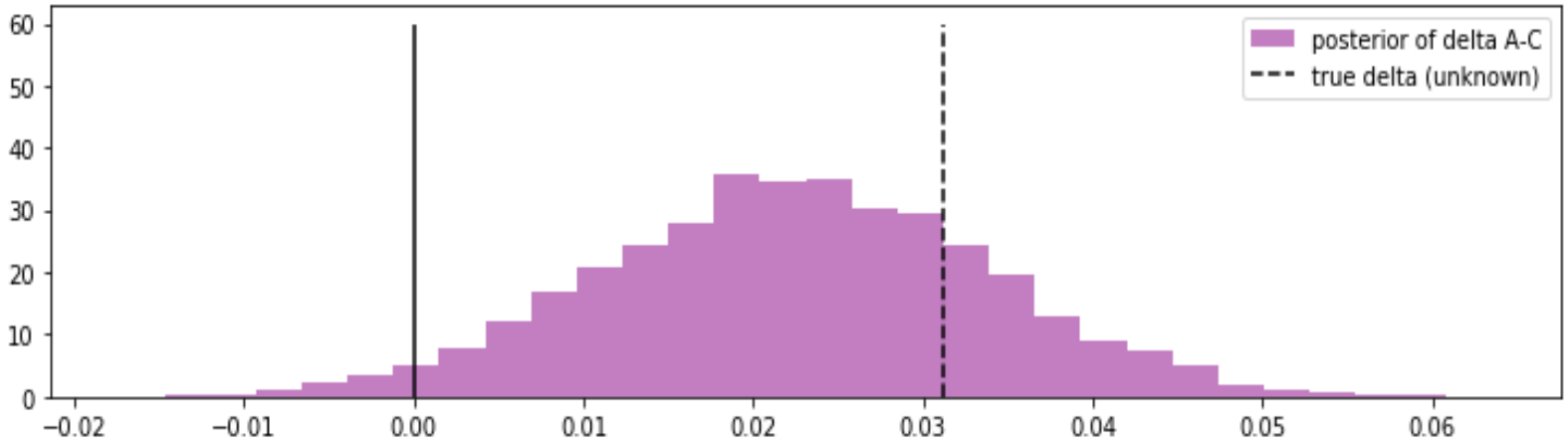


Gunadarma UG University

Analyzing Test Results

In [12]:

```
ax = plt.subplot(313)
plt.hist(delta_A_C_samples, histtype='stepfilled', bins=30, alpha=0.85,
         label="posterior of delta A-C", color="#B868B6", normed=True)
plt.vlines(true_p_A - true_p_C, 0, 60, linestyle="--",
           label="true delta (unknown)")
plt.vlines(0, 0, 60, color="black")
plt.legend(loc="upper right");
```

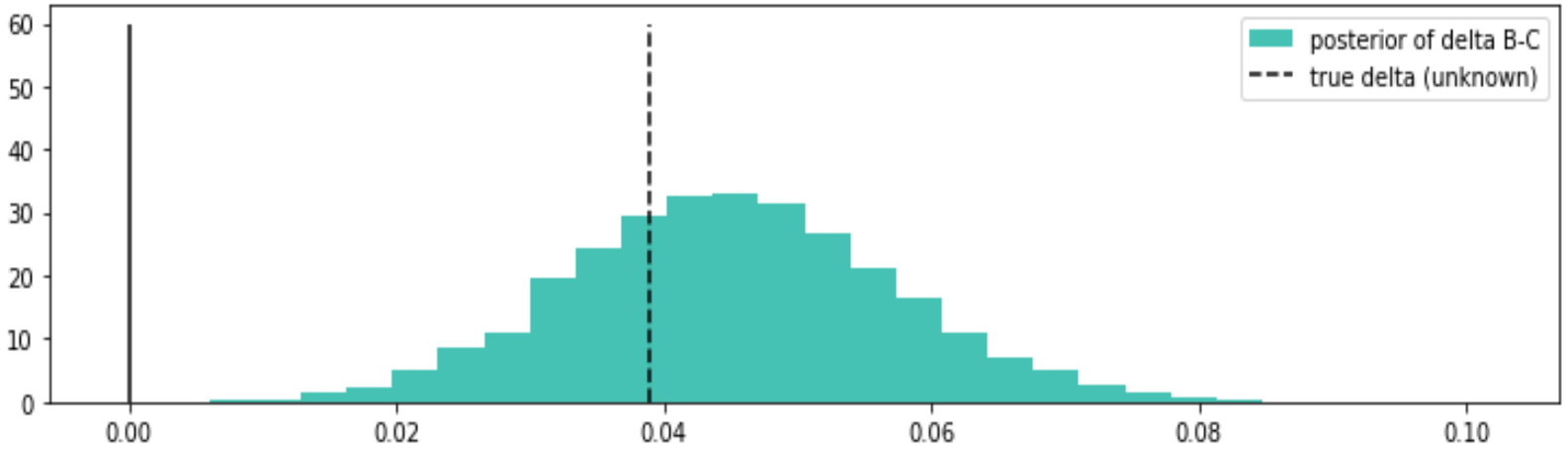


RESULT AND DISCUSSION

Gunadarma UG University

Analyzing Test Results

In [13]:
```
ax = plt.subplot(313)
plt.hist(delta_B_C_samples, histtype='stepfilled', bins=30, alpha=0.85,
         label="posterior of delta B-C", color="#24B8A6", normed=True)
plt.vlines(true_p_B - true_p_C, 0, 60, linestyle="--",
           label="true delta (unknown)")
plt.vlines(0, 0, 60, color="black")
plt.legend(loc="upper right");
```



RESULT AND DISCUSSION

Gunadarma University

Analyzing Test Results

In [14]:
```
# Count the number of samples less than 0, i.e. the area under the curve
# before 0, represent the probability that site A is worse than site B.
print("Probability site A is WORSE than site B: %.3f" % \
    np.mean(delta_A_B_samples < 0))

print("Probability site A is BETTER than site B: %.3f" % \
    np.mean(delta_A_B_samples > 0))

print("\nProbability site A is WORSE than site C: %.3f" % \
    np.mean(delta_A_C_samples < 0))

print("Probability site A is BETTER than site C: %.3f \n" % \
    np.mean(delta_A_C_samples > 0))

print("Probability site B is WORSE than site C: %.3f" % \
    np.mean(delta_B_C_samples < 0))

print("Probability site B is BETTER than site C: %.3f" % \
    np.mean(delta_B_C_samples > 0))
```

```
Probability site A is WORSE than site B: 0.961
Probability site A is BETTER than site B: 0.039

Probability site A is WORSE than site C: 0.025
Probability site A is BETTER than site C: 0.975

Probability site B is WORSE than site C: 0.000
Probability site B is BETTER than site C: 1.000
```

CONCLUSION

After performing A / B testing on the website mekargo.id it can be concluded that the final result of all combinations of probabilities generated by **version B is the best** result because it is always better when compared with versions A and C.

More Information
GUNADARMA UNIVERSITY
Jl. Margonda Raya 100, Pondok Cina - Depok, Indonesia
Telp. (+62-21) 7888 1112

CONCLUSION REMARKS

SUGGESTION

Future research can be attempted to perform A / B testing with more versions and use another method of bayesian inference method with multi-arm bandit algorithm for more complex problem.

CONCLUSION REMARKS

Thank You

More Information
GUNADARMA UNIVERSITY
Jl. Margonda Raya 100, Pondok Cina - Depok, Indonesia
Telp. (+62-21) 7888 1112